

一、变分自编码器生成 MNIST 手写数字（结合代码描述实现步骤以及提交下面要求提交的结果）

推荐使用以下函数初始化参数，可以避免一部分模式坍塌问题。

```
def glorot_init(shape):  
    return tf.random_normal(shape=shape, stddev=1. / tf.sqrt(shape[0] / 2.))
```

1、模型架构：

① 编码器（全连接层）：

输入图片维度：784 (28×28)

隐藏层维度（ReLU）：256

输出层维度（Tanh）：512

② 生成均值（全连接层）：

输入层维度：512

输出层维度：2

③ 生成标准差（全连接层）：

输入层维度：512

输出层维度：2

④ 使用均值和标准差生成隐变量z

⑤ 解码器（全连接层）：

输入维度：2

隐藏层维度（ReLU）：512

输出层维度（Sigmoid）：784

训练完网络，需要提交重构损失和KL散度的随迭代次数的变化图，以及10张生成的手写数字图片。

二、循环神经网络用于多变量时间序列预测任务（结合代码描述实现步骤以及提交下面要求提交的结果）

1. 数据集说明：北京PM2.5数据集。通过前2个小时的污染数据和天气条件预测当前时刻的PM 2.5污染。共43824条数据。

<https://archive.ics.uci.edu/ml/datasets/Beijing+PM2.5+Data>

2. 数据集示例：

No	year	month	day	hour	pm2.5	DEWP	TEMP	PRES	cbwd	lws	ls	lr
25	2010	1	2	0	129	-16	-4	1020	SE	1.79	0	0
26	2010	1	2	1	148	-15	-4	1020	SE	2.68	0	0
27	2010	1	2	2	159	-11	-5	1021	SE	3.57	0	0
28	2010	1	2	3	181	-7	-5	1022	SE	5.36	1	0
29	2010	1	2	4	138	-7	-5	1022	SE	6.25	2	0
30	2010	1	2	5	109	-7	-6	1022	SE	7.14	3	0

3. 数据预处理说明：

后续会提供一份预处理代码供参考，主要为缺失值处理、数据集划分等基础代码。数据预处理方式不做硬性要求。

4. 模型说明：

输入维度为前两个小时共16项的污染数据，构造隐层单元数为100的循环神经网络RNN，再接入输出维度为1的全连接层，预测当前PM 2.5的值。

损失函数可选择MAE，优化器可选择SGD或Adam。

5. 提交说明：

提交训练集和测试集的损失随迭代次数的变化图，训练结束后挑选2个测试集中的例子进行预测与展示。

6. 加分项：数据集分析与更好的预处理方式，不同的实验设置如不同的优化器选择，引入Dropout机制防止过拟合等等。

三、实验注意事项

1. 本作业提供上机GPU实验环境，以上两个作业至少一个使用GPU环境完成，并在对应作业说明代码实现上是否有所不同，并简单展示GPU环境相对于CPU环境的加速情况。

2. 实验室环境如下：

实验地点：B3-451

主机配置：Windows 10, RTX 2080 8G显卡, CUDA 10.1。

深度学习环境：Tensorflow-1.14.0 (Keras), Tensorflow-2.0.1 (Keras), PyTorch-1.2.0。均为Python 3.6版本。

编程软件：PyCharm, VSCode, Anaconda相关套件。

3. 实验室资源有限，需要分为5到6批上机完成实验。时间初步定在第九周和第十周，具体时间待定。

4. 建议同学们预先写好基本的代码结构并进行简单测试，尽量在实验课按时完成。